Enhance learning with ITS style interactions between learner and content

Xiangen Hu, Trey Martindale
Workforce Advanced Distributed Learning Co-Lab
The University of Memphis
xhu@memphis.edu, emartndl@memphis.edu

ABSTRACT

Much e-learning content has been produced and is being delivered as uninspiring page-turners. Although advanced learning technologies such as Intelligent Tutoring Systems (ITSs) have been shown to produce significant learning gains, it is prohibitively expensive to convert existing e-learning content into more interactive learning environments. In this paper we describe a process that may produce greater learning gains with existing e-learning content with minimal conversion time and expense. We call this ITS-enhanced delivery of shared content objects (SCOs). This process was developed by the research associates of the Workforce ADL Co-Lab at the University of Memphis. It is based on years of extensive research and development in cognitive learning theory, human tutoring, ITSs, and other advanced learning systems. The prototype we will present is supported by a contract from the Joint ADL Co-Lab.

The core of this process is a lightweight natural language processing (NLP) component that can be added to any SCO. In this process, the following scenario occurs: A student is participating in page-turning instruction. The learning management system (LMS) asks the student a question about the content. The NLP component understands the students response and offers meaningful feedback. The LMS requires the student to reflect, explain, or otherwise spend more time with the content. The resulting enhanced instructional content is a SCO that can be delivered in any SCORM-conformant LMS. The pedagogical foundation guiding the interaction between the student and the LMS is based on analysis of hundreds of hours of human tutoring and numerous studies of effective ITSs (including AutoTutor, developed by our Workforce ADL Co-lab research associates). Our paper will describe implementation of the NLP component, communication between API and LMS, and the feedback process for the student. We will demonstrate some enhanced SCOs that are used in the current Joint Knowledge Online (JKO) initiative.

ABOUT THE AUTHORS

Dr. Xiangen Hu is Associate Professor in the Department of Psychology with the University of Memphis. His training was in mathematics and computational methods (in P.R. China). His primary research interests include mathematical psychology, research design, and statistics. Dr. Hu has been PI, Co-PI, and senior researcher for numerous NSF and ONR grants. His research, development, and service activities include computational linguistics (semantic analysis), Intelligent Tutoring Systems (standardization of ITS), and Workforce ADL Co-Lab Director). He launched SOLE (summer online learning experience), which reaches 170 schools in Memphis and Shelby County every summer.

Dr. Trey Martindale is Associate Professor and Program Coordinator for Instructional Design and Technology degree programs with the University of Memphis. He is a research scientist with the University of Memphis Institute for Intelligent Systems (IIS)–researchers creating intelligent tutoring and instructional systems of the future. Dr. Martindale is the Associate Director of the Workforce Advanced Distributed Learning (ADL) Co-Lab. His research and expertise is in the design and analysis of online learning environments, and effective pedagogical use of social software tools for enhancing collaborative learning environments. His research efforts have been funded by the Centers for Disease Control and Prevention, the U.S. Department of Education, the U.S. Department of Defense, and Microsoft, among others. More information can be found at http://teachable.org.

Enhance learning with ITS style interactions between student and content

Xiangen Hu, Trey Martindale
Workforce Advanced Distributed Learning Co-Lab
The University of Memphis
xhu@memphis.edu, emartndl@memphis.edu

INTRODUCTION

E-learning has benefited from the development of standardized formats for the delivery and use of learning objects. One such standard, the Sharable Content Object Reference Model(SCORM, 2004), was developed to address the need for interoperability of learning objects within and between learning management systems. SCORM provides a standard format for creating learning objects which can be easily transferred from system to system. SCORM is a collection of specifications adapted from multiple sources that enable interoperability, accessibility and reuse of Web-based instructional content. Following the SCORM standard, instructional designers and developers can produce instructional content for learning management systems much more rapidly.

SCORM is a technological enabler of delivering learning opportunities any time, anywhere, under any context, on any system. However, SCORM is not a specification for the way instruction is presented, nor does it insure the quality of the instruction, or efficacy of learning.

Despite its advantages, SCORM has not enabled significant progress toward developing standards for more advanced learning environments such as intelligent tutoring systems (ITSs). Part of the reason for this is that SCORM specifies fixed packages of content. By contrast, content used in ITSs tends to be structured to enable flexible delivery in response to moment-by-moment actions of the student. In an ITS, delivery and content are yoked. Thus SCORM is difficult to apply in an advanced learning environment such as an ITS, and certainly does not provide a standard method for developing ITS content. Because of the many documented benefits of ITS (anderson, Corbett, Koedinger & Pelletier, 1995; Chi, Siler, Jeong, Yamauchi & Hausmann, 2001), the gap between SCORM standards and ITS is one that the developers of SCORM (the Advanced Distributed Learning Co-Lab, or ADL), ultimately intend to bridge(SCORM, 2004). The ADL vision for SCORM is that as the available inventory of SCORM objects grows, relevant objects can be assembled and used by an ITS as an intelligent, real-time interactive lesson that includes natural language interaction.

Presently the e-learning field is awash with non-interactive page-turning learning environments(Person, O'Brien, Flinn & Archer, 2005). These learning environments are less than optimal. For example, these learning environments are non-adaptive, with minimal or no feedback provided to the student. The most effective learning environment is one-on-one expert human tutoring(Chi et al., 2001; Person et al., 2005; Graesser, Ventura, Jackson, Mueller, Hu & Person, 2003; Graesser & Person, 1994), which is difficult to replicate, even within an ITS. Modifying an effective ITS by making it domain-independent and capable of using reusable content on its own is an even more daunting task. In this paper we propose a method by which we can begin to emulate the nature of an advanced ITS while using standard SCORM content.

TOWARD THE IDEAL LEARNING SYSTEM

Constructivist theories of learning emphasize the importance of the student actively constructing explanations(Chi et al., 2001; Graesser, Wiemer-Hastings, Kreuz, Wiemer-Hastings & Marquis, 2000). Researchers have developed intelligent tutoring systems that adaptively respond to a students knowledge and help construct explanations(anderson et al., 1995; VanLehn, Siler, Murray, Yamauchi & Baggett, 2003). Empirical research in discourse processing has documented the collaborative, constructive activities that frequently occur during human tutoring.

It is well documented that deeper processing leads to more robust learning(Craik & Lockhart, 1972). Students that spend more time thinking about a topic will perform better on subsequent tests on the topic(Person et al., 2005). Students that form connections between the instructional topic and other, similar topics also show better recall(Craik & Lockhart, 1972). Consequently, a primary goal in instruction is encouraging students to elaborate on the instructional content, process it deeply, and relate it to pre-existing knowledge structures.

One-on-one human tutoring produces substantial learning gains compared to other forms of learning, such as classrooms and reading from a text(anderson et al., 1995; Person et al., 2005; Graesser, VanLehn, Rose, P. & Harter, 2001). One of the reasons for the success of one-on-one human tutoring is the opportunity for the student to elaborate on the instructional content through conversation. Page-turning software, whether using SCORM objects or not, do not match the learning gains enabled by one-on-one human tutoring.

A SIMPLIFIED INTELLIGENT TUTORING SYSTEM

In the typical human tutoring scenario, a student is paired with a person who is an expert in both the content to be learned, and in pedagogical techniques. The student attempts to solve problems or answer questions about the content. The tutor evaluates the student's work in real time, and gives appropriate feedback to help the student learn. The tutor's responses may be in the form of prompts, hints, challenges, paraphrasing, encouragement, and other forms of interaction(Graesser, Wiemer-Hastings, Wiemer-Hastings, Kreuz & the Tutoring Research Group., 1999).

For example, a typical dialog between tutor and student may be as simple as the following(Graesser et al., 1999):
Tutor's Question: *Suppose a runner is running in a straight line at a constant speed, while carrying a football. While running, the runner throws the football straight up, over his head. Where will the football land? Explain why.*
Expected Answer: The football will land in the runner's hands.
Following the question from the tutor, the student and tutor interact in an alternating turns. The student responds to the tutor's questions and modifies his or her answers based on the tutor's feedback:

1. *I think–correct me if I am wrong, it will land back in the runner's hands.*
2. Positive feedback from the tutor
3. *The reason is clear; the runner and the ball have the same horizontal speed.*
4. Positive feedback from the tutor
5. *The football will land in the runner's hands.*
6. ...

Such interactions between tutor and student have been implemented and analyzed via AutoTutor, an ITS created by researchers with the University of Memphis(Franceschetti, Karnavat, Marineau, McCallie, Olde, Terry & Graesser, 2001). The researchers found that questioning a student about instructional content is an effective way of encouraging deep processing and promoting self-monitoring–knowing whether the content has in fact been learned. Students learn significantly better using AutoTutor than reading non-interactive content when AutoTutor was used to tutor students in qualitative physics (Graesser et al., 2001; Franceschetti et al., 2001).

Our goal in this paper is to describe an ITS that enables interactive dialog between the student and a virtual tutor, so the tutor will help the student engage in deeper cognitive processing about the instructional content during the interaction. We call this *Intelligent delivery of SCOs* (sharable content objects). A SCO (as defined by the SCORM standard) is an independent object that can be delivered as instructional content within a learning environment.

INTELLIGENT DELIVERY OF SCOS

Is there a way to incorporate existing SCOs that are often used in page-turning products to build an intelligent and more interactive learning environment with the effectiveness of an ITS? We believe that the most effective approach to bridging the gap between SCORM content and ITSs is via an intermediate step–enhancing existing SCORM content. We describe this process as content enhancement. Along with bridging this gap, a content enhancement system has educational benefits in its own right.

Enhancing SCORM content means that the student will have access to supplementary material, previously learned content, related topics, summaries, evaluative questions, and answers to content questions. Content enhancement means that for each "page" of content, the student will spend more time reading and thinking about the content, resulting in learning gainsmuch like a human tutor inspires cognitive disequilibrium (Graesser & Olde, 2003; Graesser, Hu & McNamara, 2005) which then requires increased student cognition.

We do not propose at this stage to describe a method for building a SCORM-conformant ITS. Instead, we have developed a system for elaborating upon SCORM content in ways that facilitate learning, and that will ultimately be usable by an ITS.

We believe the existing content of early and present day learning environments can be leveraged in advanced learning management systems that enhance the delivery of this content and increase the amount of learning that occurs. Using existing SCOs in an intelligent system can enhance learning much like an ITS. There are some basic requirements for this approach:

No SCOs Conversion: Instead of converting SCOs in SCORM 1.2 courses to SCOs that are compatible with SCORM 2004 courses (where possible adaptive learning can be achieved), we propose to use SCOs in their original form and deliver them in an enhanced environment.

No new LMS will be created: We do not propose to build an LMS (learning management system) that will have the enhanced delivery features. We instead propose to provide utilities that can be used by existing SCORM-compatible LMSs.

No new information will be added to SCOs: We propose the use of existing metadata for the SCOs and available text information (content) in the raw data of a SCO.

We believe that the most constructive approach that satisfies the above three basic requirements is to

Maximum use of existing information: Make the best use of existing metadata and raw media of SCOs

Use existing computational linguistics tools: use computational linguistics tools to help the student understand the content in the SCO.

We call this process content enhancement. Implementation of such enhancements are based on learning theories developed in cognitive psychology.

Such content enhancement means that for one page of content, the student will spend significantly more time reading, thinking, and cognitively digesting the content, hopefully resulting in significant learning gains. This process is analogous to the way a human tutor inspires cognitive disequilibrium which necessitates more cognitive activity by the student.

Semantic representation of leaning content

To enable the student's interaction with the instructional content mediated by the tutor, the tutor needs to **understand** the student's contributions in order to offer **appropriate feedback**. Understanding the student's contribution, especially in natural language, requires semantic understanding of the content. We will use a semantic engine to encode the instructional content (metadata, textual information presented, and expected answers to the tutor's questions). In this report, we used Latent Semantic Analysis (LSA) as our semantic engine(Deerwester, Dumais, Furnas, Landauer & Harshman, 1990; Dumais, Furnas & Landauer, 1988; Landauer & Dumais, 1997; Graesser, Penumatsa, Ventura, Cai & Hu, 2007; Hu, Cai, Wiemer-Hasting, Graesser & McNamara, 2007).

Latent Semantic Analysis (LSA) LSA is one the most common techniques for the encoding and representation of semantic information for a given body of written text. LSA is a generalization of familiar representations such as keyword, extended keywords method(Hu et al., 2007). Creating an LSA "space" requires extensive computation, but the steps involved are actually quite simple (Deerwester et al., 1990; Landauer & Dumais, 1997; Hu, Cai, Graesser, Louwerse, Olney, Penumatsa & TRG, 2003; Hu, Cai, Franceschetti, Penumatsa, Graesser, Louwerse, McNamara & TRG, 2003; Hu et al., 2007).

Data Acquisition: First we collect a large body digitized of texts in their natural structure (keeping the original organization, such as a book, chapter, section, paragraph, sentence, phrase, or word). Assume we have M paragraphs and N words. We then create a so-called word-document matrix where the columns are indexed paragraphs and rows are indexed words

$$\mathbf{A} = \left(f_{ij} \times G(i) \times L(i,j)\right)_{i,j}.$$

where $i \leq N$, $j \leq M$and Each entry (i,j) is a function of the frequency of the word i in the paragraph j, and some measures of importance of word i, $G(i)$, and importance of word i in paragraph j, $L(i,j)$.

Singular Value Decomposition (SVD): SVD decomposes the $\mathbf{A}$ three matrices $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}}$, where $\mathbf{U}$ has the same number of rows as the words, and $\mathbf{V}$ has the same number of columns as the number of paragraphs.

Dimension Reduction: Instead of using the all the dimensions in $\mathbf{U}$, which is the number of diagonal elements of $\mathbf{\Sigma}$, we only use the first K dimensions $(300 \leq K \leq 500)$ in $\mathbf{U}$. We denote it as $\mathbf{U}_K$. Each word is represented as K dimensional vector. We call it an LSA vector (or LSA index), which represent the "semantics" of the corresponding word.

LSA is only one of many methods one can use to represent semantics. In general, semantics are represented by numerical vectors and also use some algebraic operations to find "semantic" relations. In the case of LSA, the semantic relations between words is measured by simple normalized dot-product, between two word vectors, which is called *cosine match*. Semantic representation of words can be extended to phrases, sentences, even documents. In LSA, semantic representation of phrases, sentences, or any collection of words can be represented by the algebraic sum of the semantic vectors of the words involved. In the same fashion, the semantic relations between phrases, paragraphs, etc. can be obtained just as it is for words because they have the same vector representation.

STUDENT'S CHARACTERISTICS CURVES (LCC)

LSA has been widely used in information retrieval(Dumais et al., 1988; Graesser, Hu, Person, Jackson & Toth, 2002), similarity measurements between texts(Hu et al., 2007), text cohesion analysis(Foltz, Kintsch & Landauer, 1998), and intelligent tutoring systems (ITS)(Graesser, Wiemer-Hastings, Wiemer-Hastings, Harter, Person & the TRG., 2000; Graesser, Hu, Olde, Ventura, Olney, Louwerse, Franceschetti & Person, 2002). In this report, we provide a way to use LSA (or other semantic representations) to enhance the delivery of instructional content. The idea is as follows:

1. Represent instructional content (metadata from SCOs, text from the instructional content, questions, and expected answers) in the form of semantic vectors.
2. Encode the student's contribution using the same semantic engine.
3. Compute the similarity between the student's input and the stored answers
4. Offer feedback so the student knows if the response to the tutor's question was relevant to the answer

The entire process can be understood in terms of keyword matching(Hu et al., 2007). In fact, keyword matching inspired the approach we present here. Consider a case where a student responds to a tutor's question in multiple turns. In other words, the student could not provide the complete answer in a single response. What would be a reasonable way that a human tutor would react to a sequence of incomplete answers?

	Old	New
Relevant	O-R	N-R
irrelevant	O-IR	N-IR

Table 1: Decomposition of student's contribution sequence. N-R: New & relevant; N-IR: New & Irrelevant; O-R: Old & Relevant; and O-IR: Old & Irrelevant.

If we denote the sequence of contributions as s_i, $i = 1, ..., I$, for every contribution, the tutor would give feedback based on the four different types of information (See Table 1) for each contribution from student.

- Relevant to the answer
  - New contribution
  - Old contribution
- Irrelevant to the answer
  - New contribution
  - Old contribution

It is understandable that a human tutor would offer positive feedback when a student is providing new and relevant (N-R) contribution. Furthermore, if a student is actively constructing relevant answers, one would see a non-decrease value for the cell (N-R) in a sequence of responses. In the same fashion, other cells can be used as an indication of a student's knowledge. For example, an increasing value for the (N-IR) would indicate the lack of relevant knowledge. Consider the characteristics of the four cells in Table 1. We call them the student's characteristics curves (LCC).

One of the challenges of an ITS is to create a student model(Graesser, Person, Harter & TRG, 2001) –namely, try to assess a student's knowledge. For example, an experienced human tutor can estimate how much a student knows or does not know by evaluating a student's answers to key questions. Furthermore, a human tutor can provide feedback to help a student actively construct responses that are relevant to the questions asked. We believe the LCC is a solution that allows an ITS to create a student model and offer appropriate feedback.

Given that there are semantic vectors for answers $\mathbf{a}$, and $\mathbf{s}_i$, $i = 1, ..., (k-1)$, the k^{th} value of the LCC can be constructed in the following steps:

1. decompose $\mathbf{s}_k$ into two vectors $\mathbf{s}_{k,1}, \mathbf{s}_{k,2}$:
   $\mathbf{s}_{k,1}$ parallel to the answer vector, contains *relevant* information.
   $\mathbf{s}_{k,2}$ perpendicular to the answer, contains *irrelevant* information.
2. create spanned subspace from previous L responses, $\mathbf{s}_i$, $i = (k-L), ..., (k-1)$. $L = 0$ means no previous response is considered. Denote it as $\mathcal{S}_{L,i}$.
3. decompose $\mathbf{s}_{k,1}$ into two vectors:
   - $\mathbf{s}_{k,1,1}$ is the projection of $\mathbf{s}_{k,1}$ to subspace $\mathcal{S}_{L,i}$, $\mathbf{s}_{k,1,2}$ is projection of $\mathbf{s}_{k,1}$ to the norm of norm $\mathcal{S}_{L,i}$.
   - $\mathbf{s}_{k,2,1}$ is the projection of $\mathbf{s}_{k,2}$ to subspace $\mathcal{S}_{L,i}$, $\mathbf{s}_{k,2,2}$ is projection of $\mathbf{s}_{k,2}$ to the norm of norm $\mathcal{S}_{L,i}$.

Following steps 1 – 3, LCC values for the k^{th} response can be obtained (Table 2).

	Old	New
Relevant	$cos\,(\mathbf{s}_k, \mathbf{s}_{k,1,1})$	$cos\,(\mathbf{s}_k, \mathbf{s}_{k,2,1})$
irrelevant	$cos\,(\mathbf{s}_k, \mathbf{s}_{k,1,2})$	$cos\,(\mathbf{s}_k, \mathbf{s}_{k,2,2})$

Table 2: Computation of k^{th} values for LCC when semantic space used is LSA.

APPLICATION OF LCC IN ELEARNING

As seen in the previous section, LCC curves are determined by several factors:

1. Knowledge representation: Domain and encoding method of the semantic space
2. Content information: Answers to the tutor's question
3. Computation parameters: There are several parameters, such as the number of previous responses in the student's response history.

From items 2 and 3 above, LCC may not be used as a general purpose student model. Instead, LCC can only be used as a context dependent student model. For this reason, simulations are used to create LCC based on preset parameters.

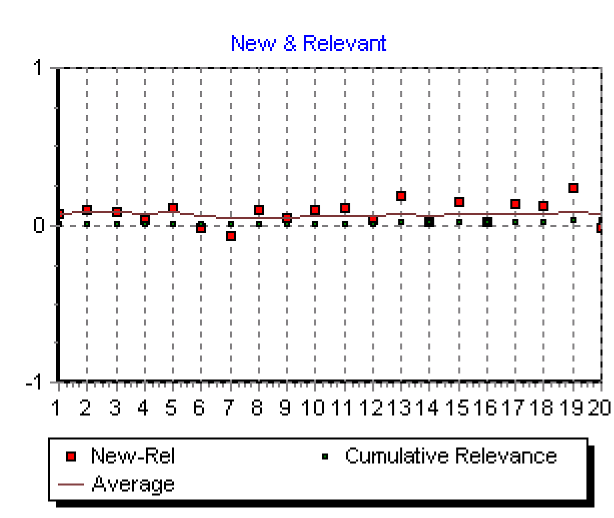

Figure 1: $p = 0.1$

Simulate LCC

To illustrate the simulation, consider the following simplified situation that specifies an interactive learning environment that involves a student and a tutor:

- G possible glossary terms in a given course
- A glossary terms contained in an answer key for a given question
- r: number of glossary terms that appear in each of the students' responses
- In each response, there are probability p that a glossary term will be in the answer key

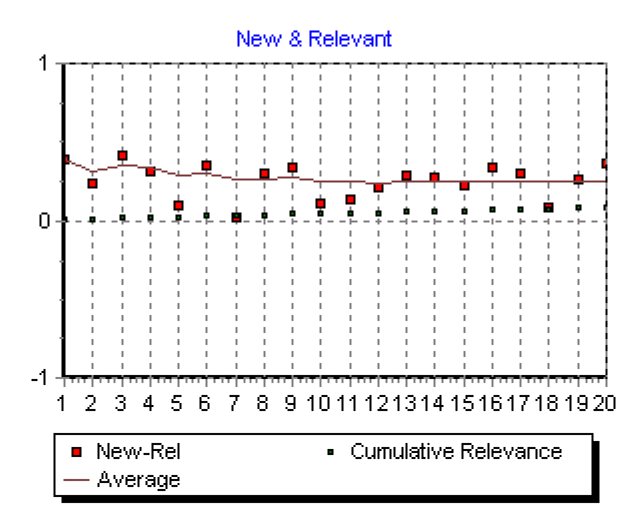

Figure 2: $p = 0.5$

Obviously, this is a simplified scenario compared to the complicated interaction that occurs during student learning. However, this scenario will approximate certain characteristics of a student's ability and the domain. For example, R and p combined could be used to describe how much *relevant information* each response contains. Using the algorithm described above, we are able to simulate a LCC for any combination of parameter values. For example, Figure 1, 2, and 3 are simulated results for a sequence of 20 responses where the number of terms in G is 1000, the size of ideal answer is 40,the history size is 2, and the size of the response is 10 with variable p values.

LCC in interactive dialog

It is worth noting that the computation of LCC is relatively simple. Usually, semantic processing requires fairly extensive computing power. For example, generating LSA requires SVD for large sparse matrix $\mathbf{A}$. However, the LCC only uses ready-made semantic vectors. The decomposing of vectors can be easily done at the client level with a java applet or via ActionScript (Adobe Flash). In applying LCC in e-learning, LCC for a given content is usually simulated. To use LCC values to evaluate a student's response, we first compute the semantic similarity values between the student's response and the answer key. Then compare the similarity values with the simulated LCC values. To illustrate how LCC can be used, the following is a description of an application that has been implemented at the Workforce ADL Co-Lab. This is a very simple use of LCC where the semantic vectors are orthogonal to each other (so it is equivalent to keyword matching, instead of semantic

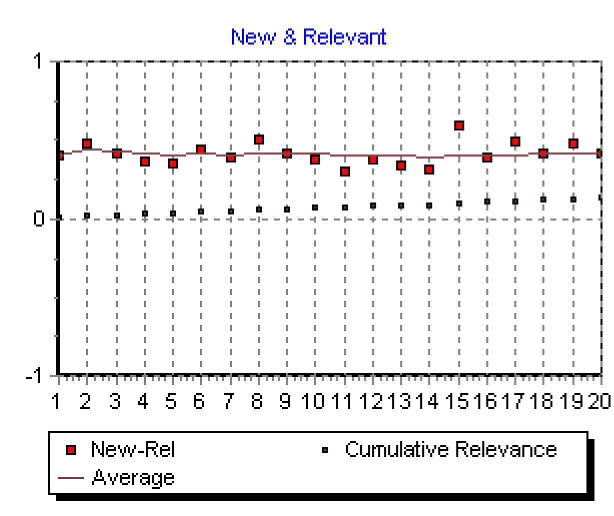

Figure 3: $p = 0.9$

similarity).

Initial implementation A SCO was produced to teach Newtonian Physics in a online qualitative physics course. The instructional strategy entails enabling a conversation between the student and the ITS. The ITS begins by posing the seed question: *Suppose a runner is running in a straight line at constant speed, and the runner throws a pumpkin straight up. Where will the pumpkin land? Explain why.*, and a sequence of short questions. We were able to have the student interact with the SCO in a conversational style (See Figure 4).

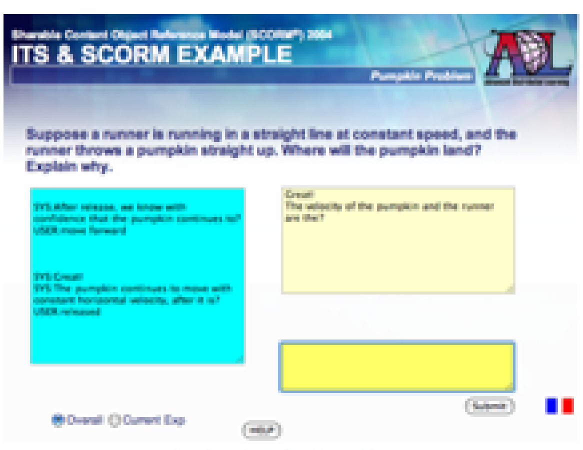

Figure 4: Example Implementation of LCC. Packaged SCOs available upon request.

Current project In this section we describe our project to produce an ITS-enhanced SCO for the Joint Knowledge Development and Distribution Capability (JKDDC). Our goal was to enable student interaction with online content in a meaningful fashion in that the student considered and then constructed answers to a seed question. A typical question would be similar to this: *What thoughts occurred to you as you watched the video?* The student would then respond in kind with a typed answer.

In order to use LCC as a student model to evaluate a student's response, these are the following steps:

Step 1: Prepare for simulation:

- **Identify a domain:** Collect domain knowledge and adapt or create a semantic space for the domain.
- **Encode course terms (G):** For the given semantic space, obtain semantic vectors of all the glossary terms in the course.
- **Encode ideal answer (A):** For the given semantic space, obtain semantic vectors of all the glossary terms in the ideal answer (of the seed question).

Step 2: Set simulation parameters:

- **Select knowledge parameter (p):** This is not a measure of *knowledge* but we can assume it is positively related to how much the student knows.
- **Select the size of response (r):** Best if this is similar to the number of terms in each sentence.

Step 3: Simulate and obtain LCC with the parameters specified in Step 2.

For each set of parameters, there will be a set of LCC. The set of LCC will be used in evaluating the student's response to the questions. For each response from the student, four values will be obtained (as it is in table 2). The values will be compared with the simulated LCC values with similar parameters. The evaluation of the student's knowledge would be the a monotonic function of p.

Future Plans

LCC methodology offers a promising solution in evaluating quality of a student's response to a given question. In previous sections, we described how LCC is used in evaluating responses from a question written prior to the learning content. We believe the LCC can be used in automatic fashion when combined with advanced techniques in computational linguistics and available information from course metadata. As it has been demonstrated, in order to use LCC, one needs to have the following elements (both for simulating LCC and using LCC online):

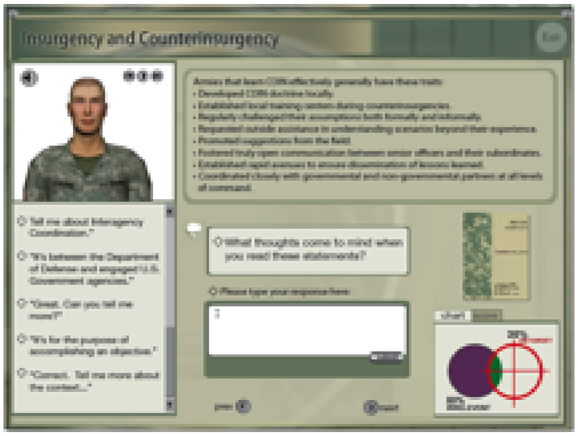

Figure 5: Example interactive SCO for Joint Knowledge Online (JKO)

Domain knowledge in the form of semantic space: This is used for encoding glossary terms in a given course. This needs to be made available.

Glossary terms in the form of semantic vectors: This is used to set a limited boundary for potential students' responses. This is usually available as part of a complete course.

Key terms in answer key: This is part of the course glossary terms that is included in the ideal answer of a question.

Our future plan is to use LCC to evaluate the quality of *automatic* interactive dialog between student and ITS. This is challenging because the *key terms in ideal answers* may not exist before the question is posed by the ITS. In fact, the answers are dependent upon the context. For this reason, there will no simulated LCC available to evaluate the student's input.

Fortunately, the technology currently exists to produce summaries, generate questions, answer questions, and find related lexical items. Examples of such systems include LSA(Landauer & Littman, 1990; Landauer, Foltz & Laham, 1998; Hu et al., 2007) and WordNet(Fellbaum, 1998). Implementation may require two further steps: 1) integrating all of these known techniques into a single content enhancement system, and 2) building a system that operates specifically on SCORM objects. Next, we list a few technologies that have potential for use in the next phase of development:

Summarization: Recent advances in computational linguistics have resulted in sophisticated summarization technologies. For example, LSA can be used to find those phrases or sentences which are most central to the meaning of the whole document, and then extract them to produce a summary. Summaries and outlines are useful to help a student preview material, and to provide an overview of the topic being covered. Automatic summaries can provide the student with a framework with which to understand the instructional content.

Note that the summarization does not need to be well-written. It can be a collection of relatively important terms. The summary can be used as *key terms in answer key* for a general question. For example, a prompt from the ITS could be as general as *Describe, in your own words, what has been explained in the last two slides.* Obviously, all the content from the last two slides can be used as an answer to the question. Summarization techniques may extract more important information from the content, and thereby aid in the creation of LCC.

Question generation Questioning a student about instructional material is an effective way of encouraging deep processing, as well as promoting self-monitoring (knowing whether the material has in fact been learned). Technology currently exists to automatically generate questions from a given text. Consequently, existing technology can be incorporated into a larger content enhancement system. This process will motivate and engage the user to think more about the content, thus aiding understanding of the content.

Research findings exist on how to generate deep reasoning questions. Based on Graesser and Person (1994), there are three distinct levels of questions:

Level 1: SIMPLE or SHALLOW:

- **Verification:** Is X true or false? Did an event occur?
- **Disjunctive:** Is X, Y, or Z the case?
- **Concept completion:** Who? What? When? Where?
- **Example:** What is an example or instance of a category?

Level 2: INTERMEDIATE

- **Feature specification** : What qualitative properties does entity X have?
- **Quantification:** What is the value of a quantitative variable? How much?
- **Definition questions:** What does X mean?
- **Comparison:** How is X similar to Y? How is X different from Y?

Level 3: COMPLEX or DEEP

- **Interpretation:** What concept or claim can be inferred from a pattern of data?
- **Causal antecedent:** Why did an event occur?

Causal consequence: What are the consequences of an event or state?

Goal orientation: What are the motives or goals behind an agents action?

Instrumental/procedural: What plan or instrument allows an agent to accomplish a goal?

Enablement: What object or resource allows an agent to accomplish a goal?

Expectation: Why did some expected event not occur?

Judgmental: What value does the answerer place on an idea or advice?

Our plan is to determine which levels of questions are most appropriate to apply LCC. Obviously, LCC may not be appropriate for questions requiring precise answers such as Verification questions in Level 1 and Quantification questions in Level 2.

Question answering There have been several question-answering (QA) and text retrieval competitions that have generated a plethora of systems for generating and finding relevant information (i.e. TREC and MUC). In a content enhancement system, QA systems will be used in somewhat differently than the typical use. Specifically, QA systems willl be used to provide relevant material, either as links or explicit suggestions, based on key sentences (which in turn are identified by summarization technologies, as described above). Elaborating material provides the opportunity for richer, deeper learning of the topic at hand.

In contrast to summarization, question answering will be useful to find answers from other parts of the domain. For example, this technique could be used to initiate context-free dialog and obtain answers from relevant course content that are not necessarily within the last learning episode.

In addition to technologies available in computational linguistics, SCORM standards make it possible to maximize the use of the technologies. The advantage of building a system that operates on SCORM objects is that SCORM has become the most widely used international standard for e-learning. For enhancing content, SCORM provides an additional advantage or tool for e-learning developers. Every SCORM object is packaged in metadata that provides information about the learning object and its content, such as topic area, subject, and so on. This metadata provides a rich source of information for generating questions, summaries, related material, and further learning. The system that we propose would take full advantage of the metadata to provide rich additional content to enhance learning.

In the SCORM 2004 release(SCORM, 2004), the content aggregation model has XML schema binding for Learning Object Metadata, Content Structure and Packaging, and Sequencing and Navigation information. This SCORM metadata describes the different components of the SCORM Content Model which includes components like SCOs. Metadata is a form of labeling that enables search and discovery of components. This metadata provides significant possibilities for enhancing and scaffolding the learning process in real time and for providing intelligent, tailored delivery. Until now, these possibilities have not been fully realized.

Metadata can be collected in catalogs, as well as directly packaged with the learning resource it describes. Learning resources that are described with metadata can be systematically searched for and retrieved for use especially for enhanced delivery. The three forms of metadata are elaborated here with uses for enhanced delivery.

Asset metadata Asset metadata is metadata that can be applied to raw media. This metadata is used to facilitate reuse and discoverability, principally during content creation, of such Assets within, for example, a content repository. Having knowledge of the assets that are presented allows an enhancement system to prompt a student with questions or ask the student to elaborate on the asset present.

The way asset metadata will be used is to extract possible semantic information for the asset. An asset could be an image illustrating some important principle. When a question is asked about an asset, the semantic information from metadata on the asset (such as a title or brief description) would be the used as ideal answer.

Content organization metadata Content Organization metadata describes simply the organization of the content. The purpose of applying content organization metadata is to make the content organization accessible (enabling discoverability) within, for example, a content repository and to provide descriptive information about the content organization. Using content organization in enhancement allows a system to formulate queries about what has previously been presented and how it relates to the current content been delivered.

SCO metadata SCO metadata provides descriptive information about the content represented in the SCO. This metadata is used to facilitate reuse and discoverability of such content within, for example, a content repository. This descriptive information would enable an enhancement system to ask general questions and locate important content being delivered.

The SCO metadata combined with textual information in the raw media is the primary source for answering context-dependent questions. Furthermore, information in SCO metadata can help determine the type of questions that are appropriate for the given content.

In general, the rich and varied information available in these kinds of metadata can be used by an ITS or e-learning system to generate content in real time on demand to increasing knowledge acquisition. Making use of meta tags enables not only efficient content categorization and retrieval, but also instructional enhancement.

SUMMARY

In this paper, we have introduced a methodology that may enhance delivery of instructional content in a distributed learning environment. Specifically, this methodology encourages and enables interactive and meaningful interaction between students and the content to be learned. The theoretical framework follows a constructivist theory of learning(Chi et al., 2001; Aleven & Koedinger, 2002). The implementation strategy is to create a student model by student's characteristics curves (LCC). Computationally, LCC is based on (vector-based) semantic representation of terms and some basic algorithms in linear algebra such as decomposing vectors to to predetermined directions (or obtaining projections of vectors to predetermined subspaces). The intuition for this methodology is from commonly used keyword matching.

Two examples are presented in the paper. The first is a simple implementation where LCC is created based on keywords (or orthogonal semantic vectors for all the terms). The second example is our current project where LCC is created based on LSA vectors. Future research and development plans are to incorporate more computational linguistics utilities, and maximize the use of SCORM metadata.

ACKNOWLEDGEMENTS

The authors would like to thanks all Workforce Advanced Distribution (ADL) Co-Lab research associates who have contributed to this project. Special thanks go to Drs. Art Graesser and David Dufty for their knowledge of cognitive psychology and computational linguistics. Mr. Zhiqiang Cai has been an important contributor to the early part of the project. We would like to thank prior students of the first author: Eric Mathews and Suresh Susarla. This work is sponsored by the support of Joint ADL Co-Lab through a Cooperative Agreement (N61339-07-2-0001) and NSF Grant (NSF SES-0616657) to the first author.

References

Aleven, V. & Koedinger, K. R. (2002). An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. *Cognitive Science*, *26*, 147–19.

anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences*, *4*, 167–207.

Chi, M. T. H., Siler, S. A., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science*, *25*, 471–533.

Craik, F. & Lockhart, R. (1972). Levels of processing. a framework for memory research. *Journal of Verbal Learning and Verbal Behavior*, *11*, 671–684.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society For Information*, *141*, 391–407.

Dumais, S., Furnas, G., & Landauer, T. (1988). Using latent semantic analysis to improve access to textual information. *Proceedings of Computer Human Interaction '88*.

Fellbaum, C. (1998). WordNet: An electronic lexical database. *MIT Press*.

Foltz, P. W., Kintsch, W., & Landauer, T. K. (1998). The measurement of textual coherence with latent semantic analysis. *Discourse Processes*, *25*, 285–307.

Franceschetti, D., Karnavat, A., Marineau, J., McCallie, G. L., Olde, B. A., Terry, B. L., & Graesser, A. C. (2001). Development of physics text corpora for latent semantic analysis. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*, (pp. 297–300). Mahwah, NJ: Erlbaum.

Graesser, A., Hu, X., Person, P., Jackson, T., & Toth, J. (2002). Modules and information retrieval facilities of the human use regulatory affairs advisor HURAA. In Driscoll, M. & Reeves, T. C. (Eds.), *Proceedings for E-Learning 2002: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, (pp. 353–360). Montreal, Canada: AACE.

Graesser, A. & Olde, B. (2003). How does one know whether a person understand s a device? the quality of the questions the person asks when the device breaks down. *Journal of Educational Psychology*, *95*, 524–536.

Graesser, A., Person, N., Harter, D., & TRG (2001). Teaching tactics and dialog in AutoTutor. *International Journal of Artificial Intelligence in Education*, *12*, 257–279.

Graesser, A., VanLehn, K., Rose, C., P., J., & Harter, D. (2001). Intelligent tutoring systems with coversational dialogue. *AI Magazine*, *22*, 39–51.

Graesser, A., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N., & the TRG. (2000). Using latent semantic analysis to evaluate the contributions of students in AutoTutor. *Interactive Learning Environments*, *8*, 128–148.

Graesser, A. C., Hu, X., & McNamara, D. (2005). Computerized learning environments that incorporate research in discourse psychology, cognitive science, and computational linguistics. In A. Healy (Ed.), *Experimental Cognitive Psychology and its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landaue* (pp. 183 – 194). Washington, D.C: American Psychological Association.

Graesser, A. C., Hu, X., Olde, B. A., Ventura, M., Olney, A., Louwerse, M., Franceschetti, D. R., & Person, N. (2002). Implementing latent semantic analysis in learning environments with conversational agents and tutorial dialog. In *Proceedings of the 24th Annual Meeting of the Cognitive Science Society*, (pp.$\tilde{3}$7). Mahwah, NJ: Erlbaum.

Graesser, A. C., Penumatsa, P., Ventura, M., Cai, Z., & Hu, X. (2007). Using LSA in AutoTutor: Learning through mixed-initiative dialogue in natural language. In M. S. D. . W. K. T. Landauer, D.S. (Ed.), *Handbook of Latent Semantic Analysis.* (pp. 243–262). Mahwah, NJ: Erlbaum.

Graesser, A. C. & Person, N. (1994). Question asking during tutoring. *American Educational Research Journal*, *31*, 104–137.

Graesser, A. C., Ventura, M., Jackson, G. T., Mueller, J., Hu, X., & Person, N. (2003). The impact of conversational navigational guides on the learning, use, and perceptions of users of a web site. In *Proceedings of AAAI Spring Symposium 2003 on Agent-Mediated Knowledge Management*, (pp. 9–14)., Palo Alto, CA. AAAI Press.

Graesser, A. C., Wiemer-Hastings, K., Kreuz, R., Wiemer-Hastings, P., & Marquis, K. (2000). QUAID: A questionnaire evaluation aid for survey methodologists. *Behavior Research Methods, Instruments, and Computers*, *32*, 254–262.

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R., & the Tutoring Research Group. (1999). AutoTutor: A simulation of a human tutor. *Journal of Cognitive Systems Research*, *1*, 35–51.

Hu, X., Cai, Z., Franceschetti, D., Penumatsa, P., Graesser, A. C., Louwerse, M., McNamara, D. S., & TRG (2003). LSA: The first dimension and dimensional weighting. In Alterman, R. & Hirsh, D. (Eds.), *Proceedings of the 25th Annual Conference of the Cognitive Science Society*, (pp. 587–592)., Boston, MA. Cognitive Science Society.

Hu, X., Cai, Z., Graesser, A. C., Louwerse, M., Olney, A., Penumatsa, P., & TRG (2003). A revised algorithm for latent semantic analysis. In Walsh, G. G. . T. (Ed.), *Proceedings of the 2003 International Joint Conference on Artificial Intelligence*, (pp. 1489–1491)., San Francisco. Morgan Kaufmann.

Hu, X., Cai, Z., Wiemer-Hasting, P., Graesser, A., & McNamara, D. (2007). Strengths, limitations, and extensions of LSA. In D. S. T. Landauer, S. D. McNamara, & W. Kintsch (Eds.), *Handbook of Latent Semantic Analysis* (pp. 401–426). Mahwah, NJ: Erlbaum.

Landauer, T. K. & Dumais, S. T. (1997). A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, *104*, 211–240.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to latent semantic analysis. *Discourse Processes*, 259–284.

Landauer, T. K. & Littman, M. L. (1990). Fully automatic cross-language document retrieval using latent semantic indexing. *In Proceedings of the 6th Annual Conferenceof the Centre for the New Oxford English Dictionaryand Text Research*, 31–38.

Person, N. K., O'Brien, M., Flinn, M., & Archer, E. (2005). An analysis of online tutoring.

SCORM (2004). Advanced distributed learning.

VanLehn, K., Siler, S., Murray, C., Yamauchi, T., & Baggett, W. (2003). Why do only some events cause learning during human tutoring. *Cognition and Instruction*, *213*, 209–249.